# Classification de séries d'images via une représentation spatio-temporelle

Mohamed Chelali, Camille Kurtz, Anne Puissant, Nicole Vincent

# Classification de séries temporelles d'images basée sur une représentation planaire spatio-temporelle

Mohamed Chelali*, Camille Kurtz*, Anne Puissant**, Nicole Vincent*

*LIPADE, Université de Paris, Paris, France
firstname.lastname@u-paris.fr
**LIVE, Université de Strasbourg, Strasbourg, France
firstname.lastname@unistra.fr

**Résumé.** Les séries temporelles d'images, telles que les séquences fonctionnelles IRM ou les séries temporelles d'images satellites (STIS), fournissent des informations précieuses pour l'analyse automatique de motifs complexes dans le temps. Un problème majeur lors de l'analyse de telles données est de considérer à la fois leurs dimensions temporelle et spatiale. Dans cet article, nous présentons une nouvelle représentation des données qui rend l'étude des séries temporelles d'images compatible avec un modèle d'apprentissage profond classique, tel que les réseaux de neurones à convolution 2D (CNN). L'approche proposée est basée sur une nouvelle représentation plane de la série temporelle d'images qui convertit les données $2D+t$ en images $2D$ sans perdre trop d'informations spatiales ou temporelles. Ce faisant, CNN peut apprendre en même temps les paramètres des filtres $2D$ impliquant des connaissances temporelles et spatiales. Les résultats préliminaires dans le domaine de la télédétection soulignent la capacité de notre approche à discriminer des classes complexes de couverture du sol (en agriculture) à partir d'une STIS.

## 1 Introduction

Les séries temporelles d'images sont quotidiennement produites par divers capteurs tels que l'IRM (imagerie fonctionnelle), les satellites, les drones ou les caméras classiques observant des classes particulières d'occupation du sol conduisant à une grande quantité d'images ($2D+t$). Dans le contexte de l'observation de la Terre, de nouvelles constellations de satellites acquièrent des images de haute résolution spatiale, spectrale et temporelle dans le monde entier. Par exemple, la constellation Sentinel-2 produit des séries temporelles d'images satellitaires (STIS) avec une durée de re-visite de 5 jours et une résolution spatiale de 10 à 20 mètres.

Parmi les applications potentielles des STIS, on peut citer la cartographie de la couverture terrestre (e.g. les zones agricoles, les zones urbaines) et l'identification de changements d'occupation des sols (e.g. l'urbanisation, la déforestation). La disponibilité croissante de ces données temporelles permet de produire et de mettre à jour des cartes précises de la couverture terrestre d'un territoire (Inglada et al., 2017). Afin de gérer efficacement l'énorme quantité

de données générée par ces nouveaux capteurs, des méthodes adaptées à l'analyse des STIS doivent être développées. Ces méthodes devraient permettre à l'utilisateur final d'obtenir des résultats satisfaisants avec un minimum de temps et d'efforts.

Un problème majeur lors de l'analyse des séries temporelles d'images est de prendre en compte simultanément les dimensions temporelle et spatiale du cube de données $2D+t$. La prise en compte simultanée de ces deux aspects peut, par exemple, faciliter la distinction entre différentes classes complexes de couverture agricole (par exemple, les vergers, les prairies) à partir des STIS. Cet article se concentre sur ce problème spécifique. Pour le traiter, nous définissons une nouvelle représentation spatio-temporelle de séries temporelles d'images qui permet de bénéficier du cadre classique de l'apprentissage profond (initialement proposé pour les images $2D$). Notre contribution principale est la proposition d'une stratégie pour représenter les données $2D+t$ sous forme d'images $2D$ sans perdre trop d'informations spatiales ou temporelles. Ce faisant, les réseaux de neurones convolutionnels (CNN) peuvent apprendre des filtres $2D$ impliquant à la fois des informations temporelles et spatiales. Ici, nous n'avons pas pour objectif de produire des cartes temporelles de la couverture terrestre ni d'étudier les changements d'occupation des sols, mais notre objectif est de cartographier des classes complexes de couverture terrestre sujettes aux confusions lorsqu'une seule image est employée.

Cet article est organisé comme suit. La section 2 rappelle certaines méthodes de l'état de l'art dédiées à l'analyse des STIS. La section 3 présente notre représentation spatio-temporelle pour l'analyse de STIS basée sur les CNN. La section 4 décrit les expériences liées à la classification de parcelles agricoles dans le domaine de la télédétection. La section 5 dresse un bilan et quelques perspectives de recherche.

## 2 Méthodes de l'état de l'art

Les STIS permettent l'observation et l'analyse de phénomènes terrestres avec une large gamme d'applications telles que l'étude de l'occupation du sol ou même la cartographie des dommages suite à une catastrophe. Ces changements peuvent être de différents types, origines et durées. Pour une étude détaillée, voir (Coppin et al., 2004).

Les méthodes pionnières d'analyse des STIS fonctionnaient sur des images simples ou des piles d'images. Sur chaque image, les différentes mesures par pixel étaient considérées comme des caractéristiques indépendantes et impliquées dans les procédures classiques basées sur l'apprentissage automatique. Dans de telles approches, la date des mesures était ignorée dans l'espace des caractéristiques. L'analyse bi-temporelle a ensuite permis de localiser et d'étudier les changements intervenant entre deux observations (Bruzzone et Prieto, 2000).

Une autre catégorie d'approches était directement conçue pour traiter les séries temporelles d'images. La plupart d'entre elles sont basées sur des approches de classification multi-dates comme l'analyse de trajectoires radiométriques (Verbesselt et al., 2010). Ces approches exploitent la notion selon laquelle la couverture du sol peut varier dans le temps (en raison des saisons, de l'évolution de la végétation (Senf et al., 2015)) et prennent en compte l'ordre des mesures à l'aide de méthodes d'analyse de séries temporelles (Bagnall et al., 2017). Chaque pixel est considéré comme une série de mesures ordonnées dans le temps (et alignées), et les modifications des mesures dans le temps sont analysées pour rechercher des motifs (temporels).

En ce qui concerne le type de caractéristiques, les approches dans le domaine fréquentiel incluent l'analyse spectrale, l'analyse d'ondelettes (Andres et al., 1994), tandis que les approches dans le domaine temporel impliquent des analyses de corrélation. En ce qui concerne la méthode de classification, la méthode classique consiste à mesurer la similarité entre un échantillon entrant et l'ensemble d'apprentissage, puis attribuer l'étiquette de la classe la plus similaire. Pour ce faire on peut utiliser, par exemple, la distance euclidienne basée sur un algorithme de plus plus proche voisin ou une méthode de distance élastique comme DTW (Petitjean et al., 2012a). Certaines méthodes proposent d'abord une projection de la STIS dans un nouvel espace, plus riche, afin d'en extraire des caractéristiques discriminantes (Petitjean et al., 2012b; Chelali et al., 2019) et la classification est réalisée dans ce nouvel espace.

Plus récemment, des approches d'apprentissage profond ont également été envisagées pour classifier les images de télédétection et générer des cartes d'occupation du sol. Dans de nombreux travaux, les réseaux de neurones convolutionnels (CNN) sont pris en compte, traitant généralement le domaine spatial des données en appliquant des convolutions $2D$ (Huang et al., 2018). Lorsqu'il s'agit de séries d'images temporelles, les convolutions sont souvent appliquées dans le domaine temporel (Pelletier et al., 2019). D'autres types d'architectures conçues pour les données temporelles sont les réseaux de neurones récurrents (RNN), comme les LSTM, utilisés avec succès dans (Ienco et al., 2017). Dans ce contexte, les approches d'apprentissage profond surpassent les algorithmes de classification traditionnels tels que les Random Forest (Ismail Fawaz et al., 2019), mais elles ne tiennent pas directement compte de la dimension spatiale des données car elles considèrent les pixels de manière indépendante. Quelques tentatives ont été réalisées pour considérer à la fois les dimensions temporelle et spatiale du cube $2D+t$ (Di Mauro et al., 2017). Une stratégie commune consiste à créer deux modèles (un pour la dimension spatiale et un pour la dimension temporelle), puis de fusionner leurs résultats au niveau de la décision. Dans le domaine de l'analyse vidéo, les caractéristiques spatio-temporelles sont apprises à l'aide de convolutions $3D$ (Tran et al., 2015), mais une telle stratégie nécessite l'apprentissage d'un nombre important de paramètres.

Dans cet article, notre stratégie consiste à classer une STIS en utilisant un modèle classique de CNN $2D$, mais nous proposons une nouvelle représentation des séries temporelles d'images intégrant simultanément les dimensions temporelle et spatiale des données. Nous proposons plusieurs représentations basées sur des stratégies variées pour prendre en compte des variations locales de pixels de manières différentes. Le CNN apprend simultanément avec des convolutions $2D$ des informations temporelles et spatiales.

# 3 Approche proposée

Cette section présente notre méthode dédiée à la classification des séries temporelles d'images basée sur une représentation planaire spatio-temporelle. Ce travail a fait l'objet d'une publication en conférence internationale (Chelali et al., 2020). Après avoir fourni une vue d'ensemble de la chaîne de traitement du processus global, nous détaillerons les différentes étapes de la méthode.

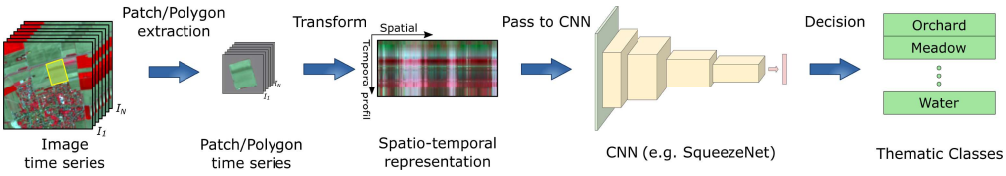


FIG. 1: Organigramme de notre méthode de classification de STIS basée sur une représentation planaire spatio-temporelle.

## 3.1 Chaîne de traitement

La méthode proposée repose sur l'utilisation d'une architecture classique de réseau de neurones profond. Mais l'entrée n'est pas une structure $3D$ comme dans (Tran et al., 2015) ni une structure $1D$ comme dans (Pelletier et al., 2019), approches fréquentes pour les méthodes de l'état de l'art qui étudient les séries temporelles associées à chaque pixel. Dans notre cas, nous proposons de considérer les pixels d'une région d'intérêt (par exemple, un patch d'image ou un polygone) dans son ensemble et d'appliquer d'abord une transformation de ces données $2D+t$ fournissant une image $2D$ (structure planaire) contenant toutes les données spatio-temporelles. Cela correspond à la partie gauche de l'organigramme présenté dans la Figure 1. Une telle structure est ensuite transférée en tant qu'entrée d'un réseau de neurones classique pour permettre la classification. Le réseau peut être conçu pour apprendre les étiquettes à partir des informations spatiales et temporelles contenues dans les données. La partie droite de l'organigramme illustré en Figure 1 illustre ce processus.

## 3.2 Représentation planaires des données : du $2D+t$ au $2D$

Afin de réduire la complexité de la structure de données, nous proposons de transformer la représentation spatiale des pixels en une structure $1D$. Initialement, un pixel est défini par sa position (un couple d'entiers) dans une image de hauteur $\mathbb{H}$ et de largeur $\mathbb{W}$. Maintenant, il sera défini par un seul entier donné par un index spécifiant la position du pixel dans un chemin couvrant la région d'intérêt. La fonction $\mathfrak{R}$

$$\begin{aligned} \mathfrak{R} : [1,\mathbb{W}] \times [1,\mathbb{H}] &\rightarrow & [1,\mathbb{W}\times\mathbb{H}] \\ (x,y) &\mapsto & i = \mathfrak{R}(x,y) \end{aligned}$$

associe à un pixel de coordonnées $(x,y)$ sa position $i$ dans un espace mono-dimensionnel.

Ce qui est important dans le plan, c'est la notion de voisinage. Un pixel a généralement 8 ou 4 voisins selon la topologie considérée. Dans une chaîne $1D$, chaque élément n'a que 2 voisins les plus proches. Ensuite, bien sûr, en transformant un espace $2D$ en un espace $1D$, les informations spatiales seront réduites, mais l'objectif est de conserver les informations les plus représentatives pendant la transformation.

Lorsqu'une transformation particulière est choisie (quelques exemples seront proposés ci-après), elle sera appliquée de la même manière à toutes les $N$ images (ou à une région d'intérêt particulière) de la série. Donc, nous obtenons $N$ chaînes qui seront considérées comme les lignes d'une nouvelle image. La nouvelle hauteur de l'image est égale au nombre $N$ des

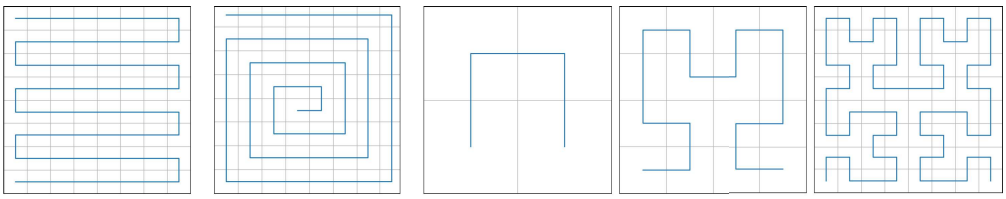


FIG. 2: Illustrations des différentes courbes (en bleu) couvrant un espace $2D$ (en noir, une grille de pixels) ; (a) Courbe serpent ; (b) Courbe spirale ; (c, d, e) Les trois premiers ordres de la courbe de Hilbert.

images de la STIS et sa largeur est égale au nombre de pixels de la région que nous voulons représenter. Cette nouvelle image constitue alors une représentation spatio-temporelle $2D$ d'une série temporelle d'images $2D+t$.

Afin de conserver certains voisins significatifs dans cette nouvelle représentation, le problème est alors de remplir un espace discret $2D$ avec une courbe discrète. En suivant les pixels le long de la courbe, tous les pixels de la région seront numérotés une seule fois et, par construction, deux pixels adjacents dans la courbe sont des pixels voisins dans le plan. Dans la littérature, de nombreuses méthodes ont été proposées pour réaliser une telle transformation, mais le but est de considérer des voisins statistiquement représentatifs sans aucun biais en raison du tracé choisi dans le plan.

Nous avons comparé expérimentalement plusieurs stratégies :

— la première représentation est la plus naïve, notée $\Re_{snake}$. L'espace est rempli par une simple courbe qui scanne l'image, ligne par ligne, en serpentant (Figure 2 (a)). Les lignes sont liées de manière intelligente, de sorte que les informations de voisinage spatial sont préservées : les extrémités des lignes impaires sont liées aux têtes des lignes paires, et vice versa. Les pixels sont alors numérotés en fonction de la courbe.

— la deuxième représentation est basée sur la spirale d'Archimède, notée $\Re_{spiral}$. La grille de pixels est associée à une courbe en spirale qui remplit un carré (Figure 2 (b)). La courbe commence à partir du point central $(0,0)$ d'un carré et de son voisin droit puis tourne autour. La construction de cette courbe se fait en fixant deux variables qui indiquent le prochain point de la courbe, $(x+dx, y+dy)$. $dx$, $dy$ sont initialisés à 0 et 1 respectivement. Les points angulaires sont ceux qui vérifient $x = y$, $x = -y$ et $y > 0$, $x-1 = -y$ et $x > 0$. La courbe doit aller à droite, à gauche, en bas ou en haut selon les directions de $(dx, dy)$. Les valeurs $(dx, dy)$ sont successivement $(0,1), (-1,0), (0,-1)$ et enfin $(1,0)$.

— la troisième représentation est basée sur des courbes de remplissage de l'espace, notée $\Re_{Hilbert}$. Notre choix est la courbe de Hilbert, qui est une courbe fractale remplissant l'espace (Butz, 1971) et qui remplit un carré (une surface $2D$). Pour définir cette courbe, un processus récursif est appliqué à partir d'un domaine carré, le domaine étant divisé en quatre carrés égaux. Les quatre petits carrés sont liés de manière à ce que deux parties avec une arête commune aient deux index consécutifs. Cette règle est appliquée de manière récursive sur les carrés dont la largeur est une puissance de 2. L'ordre des pixels est finalement donné par la courbe de Hilbert. L'intérêt principal de ce type de

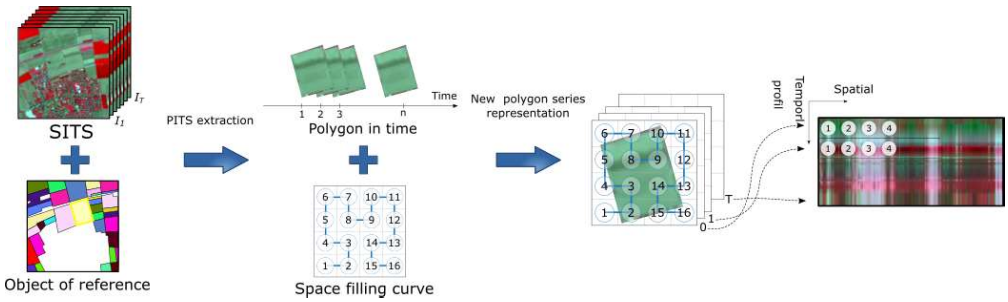


FIG. 3: Représentation d'une série temporelle de polygones basée sur la courbe de Hilbert.

courbe est la préservation de la relation de voisinage spatial de points successifs de la courbe. La Figure 2 (c–e) illustre les trois premiers ordres de courbes de Hilbert.

En appliquant le processus aux $N$ images (ou à une région d'intérêt spécifique) de la STIS, nous obtenons $N$ lignes de longueur égale au nombre $N_r$ de pixels de la région. Ces lignes sont utilisées pour remplir une matrice et une nouvelle représentation de la STIS est obtenue sous forme d'une image avec $N$ lignes et $N_r$ colonnes. Maintenant, cette nouvelle image peut également être interprétée en termes de colonnes. Chaque colonne est associée à un pixel et à sa série temporelle dans la STIS, un pixel temporel $p = \{< p_t(x,y) > | t = 1 \ldots N\}$ est contenu dans une colonne de la nouvelle image. La Figure 3 illustre la construction de la nouvelle représentation.

## 3.3 Architecture profonde employée

Les réseaux de neurones convolutionnels sont utilisés dans la plupart des méthodes appartenant à la famille des algorithmes d'apprentissage profond. Les CNN sont composés, dans la partie gauche, de couches de neurones calculant les convolutions des sorties des couches précédentes. Les neurones de chaque couche sont activés par des fonctions non linéaires permettant l'extraction de caractéristiques d'ordre élevé de l'entrée. Il existe également des couches de regroupement maximal entre les couches de convolution afin de réduire progressivement le nombre d'entrées et le nombre de paramètres à calculer pour définir le réseau et pour contrôler également le sur-apprentissage. Dans la dernière partie droite du réseau, pour résoudre les problèmes de classification, nous trouvons généralement une couche entièrement connectée fournissant un vecteur de probabilité, couplée à une fonction softmax permettant de prédire une classe.

Dans notre approche, nous considérons le modèle SqueezeNet (Iandola et al., 2016). Ce modèle est un petit réseau composé de peu de paramètres à apprendre. Dans notre cas, il s'agit d'un modèle intéressant, car il s'adapte à notre contexte applicatif et à notre jeu de données (taille réduite des exemples d'apprentissage). Ce CNN conduit au même niveau de précision que le modèle AlexNet, lorsqu'il est évalué sur le jeu de données ImageNet.


FIG. 4: Exemple de STP représentant des vergers ; (gauche) Évolution d'un verger traditionnel / intensif ; (droite) Représentations spatio-temporelles associées (ici avec la stratégie de Hilbert $\mathfrak{R}_{Hilbert}$).

# 4 Étude expérimentale

L'approche proposée a été évaluée dans le cadre d'une application de télédétection, à savoir la classification de parcelles agricoles à partir de STIS. Notre objectif est de différencier certaines classes thématiques agricoles (ici les vergers traditionnels par rapport aux vergers intensifs difficilement différentiables à l'échelle des images Sentinel-2). L'aspect visuel de ces parcelles agricoles est hétérogène car les vergers font l'objet de nombreuses pratiques agricoles, dépendant de la saison, et leur identification automatique reste une tâche complexe et importante pour différents besoins de gestion des territoires et de l'environnement. Afin de différencier ces deux classes, les caractéristiques spatio-temporelles peuvent contenir des informations utiles pour mieux discriminer les pratiques agricoles.

## 4.1 Données

Les données utilisées dans cette étude expérimentale sont des STIS optiques, captées par le satellite Sentinel-2 (Est de la France). Les données acquises ont été corrigées et orthorectifiées par le programme français Theia afin de pouvoir être comparables radiométriquement. Les images sont distribuées avec leurs masques de nuages associés. Un prétraitement a été appliqué aux images avec une interpolation linéaire sur les pixels masqués pour garantir la chérence de tous les pixels.

Nous disposons d'un STIS de $N = 50$ images capturées en 2017 sur la même zone géographique. Pour chaque image, seules trois bandes sont conservées : proche infrarouge (Nir), rouge (R) et vert (G). Toutes ces bandes ont une résolution spatiale de 10 mètres.

En plus des images, nous disposons de données de référence composées des délinéations de parcelles agricoles de référence (dans notre contexte les vergers) représentées sous forme de polygones vectoriels. Ces polygones sont extraits du RPG de l'IGN. Dans notre cas, les polygones ont été rasterisés en fonction de la résolution spatiale de chaque image, ce qui a conduit à une nouvelle série temporelle de polygones, notée STP.

Les données de référence utilisées dans notre expérience sont les étiquettes sémantiques de ces polygones (vergers traditionnels ou intensifs). La Figure 4 présente un exemple de l'évolution temporelle de deux vergers à travers la STIS. Enfin, nous disposons de 100 polygones par classe. Afin d'obtenir plus de données annotées, nous avons employé une technique d'aug-

mentation de données (AD) en appliquant des rotations sur les images avec les angles : 45°, 90°, 135° and 180°.

## 4.2 Protocole expérimental

Nous avons appliqué la méthode proposée pour classifier les deux classes de vergers (traditionnel et intensif). D'un point de vue intuitif, les vergers intensifs devraient avoir une texture plus homogène dans le domaine spatial puisque les arbres fruitiers sont généralement alignés, ce qui n'est pas toujours le cas dans les vergers traditionnels.

### 4.2.1 Préparation des données

Premièrement, les données d'entrée sont préparées grâce aux représentations spatio-temporelles d'images proposées. Ceci est opéré au niveau polygone. Chaque STP est traité de 3 manières différentes selon les fonctions $\Re_{snake,spiral,Hilbert}$ présentées précédemment. Pour souligner l'intérêt de considérer la relation spatiale entre les pixels, nous avons ajouté (comme base naïve de référence) une stratégie aléatoire pour former la représentation spatio-temporelle du STP, notée $\Re_{random}$.

En fonction de la taille d'entrée du CNN, qui est de $224 \times 224$, nous adaptons les images générées à cette taille. Pour la dimension temporelle (axe $Y$), nous proposons deux stratégies. La première consiste à centrer verticalement les informations d'origine des images d'entrée $N$ ($N = 50$). Les lignes supérieures et inférieures restantes sont fixées à la valeur zéro. Pour la seconde, nous avons choisi de traiter une série temporelle de longueur 224, c'est-à-dire de remplir tout l'espace vertical restant. Pour ce faire, nous avons appliqué une interpolation linéaire sur les informations temporelles. Nous supposons que l'information temporelle entre deux dates consécutives est monotone et linéaire. L'interpolation est ensuite effectuée en considérant que nous n'avons que 224 jours dans l'année, de sorte qu'un jour a une durée d'environ 39 heures. Pour les dates initiales, nous affectons les informations temporelles de la première date dans la STIS. Pour les dernières dates, nous affectons les dernières informations temporelles dans la STIS. Pour les autres valeurs de date inconnues, nous les calculons en appliquant une fonction linéaire qui prend en compte deux dates disponibles consécutives (prises à partir du jeu de $N = 50$ images de la STIS). Enfin, nous avons 224 dates qui complètent la hauteur de l'image. Ces deux stratégies (avec dates originales ou avec interpolation temporelle) seront évaluées séparément.

Pour la dimension spatiale (axe $X$), la taille des polygones étant rarement égale à 224, nous avons adopté la stratégie suivante. Pour les polygones dont le nombre de pixels est inférieur à 224, nous répétons la séquence. Pour ceux composés de plus de 224 pixels, nous découpons la nouvelle représentation en différentes images avec 224 colonnes, ce qui conduit potentiellement à un nombre de données à classer supérieur au nombre de polygones.

Les données images ont été normalisées en fonction des valeurs maximales et minimales du jeu de données. Dans notre cas, nous avons limité les valeurs à 2% (ou 98%), comme proposé dans (Pelletier et al., 2019).

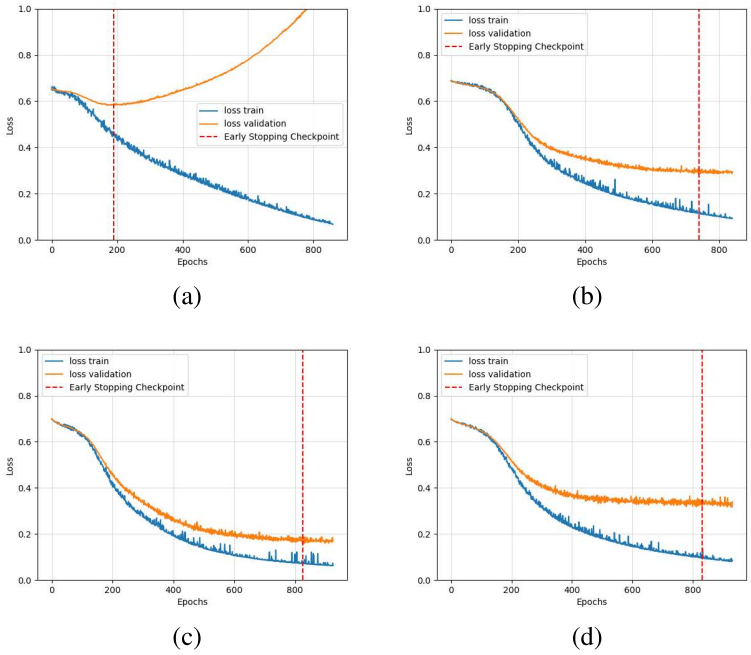


FIG. 5: Courbes de perte liées à l'entraînement de notre modèle avec les différentes représentations spatio-temporelles ; (a) stratégie aléatoire $\Re_{random}$ ; (b) stratégie serpent $\Re_{snake}$ ; (c) stratégie spirale $\Re_{spiral}$ ; et (d) stratégie Hilbert $\Re_{Hilbert}$.

### 4.2.2 Protocole d'apprentissage et de validation

Pour valider ces expériences, une stratégie de validation croisée (5 fold) est utilisée. Dans chaque cas, le jeu de données est divisé de manière aléatoire en 3 jeux, au niveau polygone, et nous répétons 5 fois le processus. La taille de ces ensembles est de 60%, 20% et 20% de toutes les données disponibles représentant respectivement les ensembles d'apprentissage, de validation et de test. Dans chaque expérience, les mêmes découpages sont pris en compte afin de rendre les résultats plus comparables. Le modèle est formé et évalué 5 fois en fonction de chaque division. Pour une division, nous considérons le système qui donne le meilleur résultat sur l'ensemble de validation. Notez que la décision de la sortie du classificateur est prise au niveau du polygone. Nous avons déjà expliqué que pour les grands polygones (qui ont plus de 224 pixels), nous construisons plusieurs images différentes dans notre processus (voir la Section 4.2.1). Ensuite, plusieurs images sont associées à un seul polygone. Pour prendre une décision dans ce cas, le modèle renvoie les probabilités de classes pour chaque image associée au polygone. Ensuite, nous faisons la moyenne de ces probabilités pour chaque classe et nous affectons au polygone l'étiquette de la classe avec la probabilité la plus élevée. Nous rapportons la précision globale qui correspond à la valeur moyenne des résultats sur les ensembles de test en fonction des 5 divisions et de l'écart type.

Nous entraînons le modèle en utilisant *Adam* comme optimiseur avec un taux d'apprentis-

sage de $10^{-6}$ et les valeurs par défaut des autres paramètres ($\beta_1 = 0.9$, $\beta_2 = 0.999$ et $\varepsilon = 10^{-8}$) avec une taille de batch de 8. Nous limitons le nombre d'époques à 2 000, à la suite d'une technique d'arrêt précoce avec un nombre de patience de 100.

La taille de l'ensemble de données disponible étant limitée, nous entraînons le réseau selon deux stratégies : (1) *from scratch* et (2) avec un *fine-tuning* (le modèle a été pré-entraîné sur ImageNet dans le cadre d'un problème de classification). Nous avons également procédé à une augmentation des données (AD).

## 4.3 Résultats et discussions

Les représentations spatio-temporelles proposées des STP ont été utilisées pour nourrir le CNN. Nous avons également utilisé l'ordre $\Re_{random}$ de pixels afin d'évaluer l'importance des informations spatiales. Deux pixels successifs dans la représentation $1D$ sont des voisins dans l'espace $2D$. Ceci est une propriété des différentes courbes de remplissage d'espace que nous avons considérées. À des fins de visualisation, la Figure 4 illustre deux STP avec leurs représentations spatio-temporelles résultantes, basées ici sur la stratégie $\Re_{Hilbert}$.

Le modèle CNN a été entraîné conformément au protocole d'apprentissage, avec et sans fine-tuning. Nous avons également évalué l'impact de la prise en compte des dates originales ou de l'application d'une interpolation temporelle pour correspondre à la taille d'entrée d'image ($224 \times 224$) requise par SqueezeNet.

La Figure 5 illustre les courbes de perte résultantes lorsque SqueezeNet est entraîné (suivant une technique d'arrêt précoce) avec des images associées respectivement aux stratégies $\Re_{random}$, $\Re_{snake}$, $\Re_{spiral}$ et $\Re_{Hilbert}$, ici avec les dates originales. À partir de ces courbes, nous remarquons que les valeurs de perte les plus élevées sont obtenues avec la stratégie $\Re_{random}$ comme prévu. De plus, la courbe de perte de la stratégie $\Re_{random}$ commence à se stabiliser dès 200 époques, par rapport aux autres stratégies qui commencent à se stabiliser à partir d'environ 600 époques. Intuitivement, cela signifie que la stratégie $\Re_{random}$ ne fournit pas une bonne représentation des STP avec une bonne capacité à généraliser lors de l'entraînement. Les autres représentations permettent de faire un meilleur entraînement. Nous pouvons également voir que les meilleures courbes d'apprentissage sont obtenues en (c), en utilisant $\Re_{spiral}$, avec les meilleurs résultats sur l'ensemble de validation. Ce classement n'est pas conservé au niveau de l'ensemble de test global.

Le Tableau 1 présente les résultats de la classification (précision globale) obtenus avec nos représentations spatio-temporelles (avec les dates originales). Nous remarquons que $\Re_{random}$ fournit toujours les scores les plus bas comparés aux autres représentations, avec et sans AD, avec et sans fine-tuning. Ceci est attendu car la discrimination entre vergers traditionnels et intensifs repose sur des informations spatiales et ces informations sont partiellement préservées avec les courbes de remplissage d'espace fournissant des informations spatiales en plus des informations temporelles. Dans le Tableau 1, nous remarquons également que, avec l'AD, tous les scores sont légèrement augmentés et que les meilleurs ont été obtenus en combinant AD et le fine-tuning. Enfin, les meilleures représentations oscillent entre $\Re_{snake}$, $\Re_{spiral}$ et $\Re_{Hilbert}$.

A titre comparatif, nous avons comparé nos résultats à ceux obtenus avec la méthode TempCNN dédiée à la classification des séries temporelles, proposée dans (Pelletier et al., 2019). Cette approche repose sur l'utilisation d'un CNN, dans lequel les convolutions sont appliquées dans le domaine temporel (convolutions $1D$). Les tailles de filtre sont fixées en fonction du critère indiqué dans (Pelletier et al., 2019) : avec une taille de noyau de 5 lors de

TAB. 1: Résultats de classification (précision globale – OA et écart type – STD) obtenus avec nos représentations spatio-temporelles (avec dates originales) ; (première / deuxième ligne) Sans / avec augmentation des données.

| | | From scratch | | Fine tuning | |
|---|---|---|---|---|---|
| | **Rep.** | **OA** | **STD** | **OA** | **STD** |
| sans AD | $\mathfrak{R}_{random}$ | 71.50 | 7.17 | 81.00 | 8.15 |
| | $\mathfrak{R}_{snake}$ | 78.00 | 4.30 | 90.50 | 7.96 |
| | $\mathfrak{R}_{spiral}$ | 76.00 | 8.74 | **92.00** | **3.31** |
| | $\mathfrak{R}_{Hilbert}$ | **79.00** | **5.61** | 91.00 | 2.00 |
| avec AD | $\mathfrak{R}_{random}$ | 80.50 | 3.67 | 87.00 | 4.58 |
| | $\mathfrak{R}_{snake}$ | 83.50 | 7.00 | **93.50** | **2.54** |
| | $\mathfrak{R}_{spiral}$ | **84.50** | **5.33** | 93.00 | 1.87 |
| | $\mathfrak{R}_{Hilbert}$ | 81.50 | 6.44 | 91.00 | 2.54 |

TAB. 2: Résultats de la classification (précision globale – OA et écart type – STD) avec les architectures TempCNN (avec les dates d'origine et un noyau de taille 5).

| nb filt. | 16 | 32 | 64 | 128 | 256 | 512 | 1024 |
|---|---|---|---|---|---|---|---|
| **OA** | 78.81 | 77.38 | 81.66 | 78.45 | **85.37** | 81.73 | 84.80 |
| **STD** | 6.08 | 6.51 | 4.59 | 4.79 | **3.44** | 5.75 | 6.48 |

la prise en compte des dates d'origine et de 11 lors de la prise en compte des dates interpolées. À des fins de comparaison, nous avons entraîné et validé le modèle TempCNN en utilisant le même protocole de validation. Dans le code proposé par les auteurs du modèle, la décision au niveau polygone est obtenue via un vote majoritaire pondéré par les probabilités issues du réseau. Notez que le modèle TempCNN est proposé avec différentes architectures (profondeurs), conduisant à un nombre différent de filtres.

Le Tableau 2 contient les résultats de TempCNN. Les meilleurs scores ont été obtenus avec 256 filtres. Les scores obtenus suggèrent que les résultats obtenus avec TempCNN surpassent ceux obtenus avec notre méthode lorsque nous entraînons le réseau à partir d'une initialisation aléatoire. Cependant, avec une initialisation du réseau avec les poids obtenus lors d'un pré-entraînant sur ImageNet, nous obtenons de meilleurs scores. Cela met en évidence, pour notre contexte applicatif, l'avantage de considérer un modèle classique de CNN $2D$ pour classifier les images $2D+t$ combinées à nos représentations spatio-temporelles.

Le Tableau 3 présente les résultats de la classification avec la stratégie d'interpolation temporelle. Nous remarquons qu'avec plus d'informations temporelles, les scores globaux sont augmentés par rapport au cas où moins d'informations temporelles (images avec dates originales) sont disponibles (Tableau 1). Cela s'explique par la distribution non régulière des dates d'origine. Avec l'interpolation, nous obtenons une information temporelle avec une régularité égale pour obtenir 224 dates. Ceci provient également du comportement monotone réel entre les dates consécutives utilisées pour l'interpolation. Nous observons à nouveau que la stratégie

TAB. 3: Résultats de classification (précision globale – OA et écart type – STD) obtenus avec nos représentations spatio-temporelles (avec interpolation temporelle) ; (première / deuxième ligne) Sans / avec augmentation des données.

| | | From scratch | | Fine tuning | |
|---|---|---|---|---|---|
| | **Rep.** | **OA** | **STD** | **OA** | **STD** |
| sans AD | $\mathfrak{R}_{random}$ | 84.00 | 9.02 | 87.00 | 4.30 |
| | $\mathfrak{R}_{snake}$ | 85.00 | 4.18 | **92.50** | **3.16** |
| | $\mathfrak{R}_{spiral}$ | 85.00 | 3.53 | 91.00 | 2.54 |
| | $\mathfrak{R}_{Hilbert}$ | **89.00** | **3.39** | 91.00 | 2.54 |
| avec AD | $\mathfrak{R}_{random}$ | 82.00 | 8.71 | 83.50 | 4.35 |
| | $\mathfrak{R}_{snake}$ | 86.50 | 5.38 | 90.50 | 1.87 |
| | $\mathfrak{R}_{spiral}$ | 86.50 | 3.00 | **91.50** | **3.74** |
| | $\mathfrak{R}_{Hilbert}$ | **92.50** | **1.58** | 89.00 | 3.39 |

TAB. 4: Résultats obtenus (précision globale – OA et écart type – STD) avec les architectures TempCNN (avec interpolation temporelle et un noyau de taille 11).

| nb filt. | 16 | 32 | 64 | 128 | 256 | 512 | 1024 |
|---|---|---|---|---|---|---|---|
| **OA** | 78.96 | 81.40 | 83.96 | 81.86 | 85.93 | 84.23 | **87.21** |
| **STD** | 7.34 | 6.32 | 7.14 | 5.18 | 8.03 | 6.23 | **8.28** |

$\mathfrak{R}_{random}$ conduit aux pires scores. Cela confirme que l'information spatiale est importante et pas seulement temporelle. Nous voyons aussi que l'AD augmente légèrement les scores en cas d'apprentissage from scratch mais n'est pas en mesure d'améliorer les résultats en cas de fine-tuning. Dans cette expérience, la stratégie $\mathfrak{R}_{Hilbert}$ conduit à la représentation qui fournit les meilleurs scores lorsque nous entraînons le réseau from scratch (avec ou sans AD). Mais lorsque nous employons le fine-tuning, les meilleures représentations oscillent entre $\mathfrak{R}_{snake}$ et $\mathfrak{R}_{spiral}$.

Les résultats obtenus avec TempCNN et la stratégie d'interpolation temporelle sont répertoriés dans le Tableau 4. Les scores initiaux sont dans le même intervalle que notre méthode lorsque nous entraînons from scratch. Mais avec AD et / ou fine-tuning, nos scores sont plus élevés.

# 5 Conclusion

Dans cet article, nous présentons une nouvelle stratégie pour transformer une série temporelle d'images en une représentation planaire spatio-temporelle. Ceci permet de réduire la complexité de la structure de la série temporelle d'images (de $2D+t$ à $2D$) tout en conservant (partiellement) les relations spatiales et temporelles des pixels. Ces représentations sont utilisées pour alimenter un CNN classique afin d'effectuer une classification. Les convolutions $2D$

peuvent alors conduire à une extraction de caractéristiques spatio-temporelles. En comparaison aux approches $1D$ dédiées aux séries temporelles, nous avons un nombre moins élevé de données annotées, mais ceci est compensé par une stratégie d'augmentation des données. En considérant des convolutions $2D$, nous pouvons également bénéficier d'un modèle pré-entraîné sur ImageNet. Une telle initialisation des poids du CNN est moins facile à réaliser pour les approches $1D$, car aucun jeu de données semblable à ImageNet n'est disponible.

L'approche proposée a été évaluée en télédétection pour la classification de parcelles agricoles à partir de STIS. Dans notre expérimentation, nous étudions l'impact de la transformation spatio-temporelle en utilisant différentes courbes de remplissage de l'espace. Les résultats obtenus reflètent l'utilité et l'impact de la prise en compte des informations spatiales et temporelles. Dans notre étude thématique, nous observons que les scores de classification sont plus élevés lorsque l'on considère les représentations spatio-temporelles avec plus d'informations temporelles (en utilisant l'interpolation temporelle) que celles qui en ont moins, même si elles sont construites à partir des mêmes données initiales. Il est donc plus important d'avoir beaucoup de données dans le domaine temporel que d'optimiser la façon dont le plan $2D$ est rempli par les courbes de remplissage de l'espace.

Dans notre étude comparative, nous remarquons que la méthode TempCNN (Pelletier et al., 2019) s'applique au niveau des pixels alors que notre approche s'applique au niveau des polygones. Cela signifie que pour TempCNN le nombre d'échantillons d'entraînement est supérieur à celui de notre méthode où les pixels d'un polygone sont tous résumés dans une seule image spatio-temporelle. Malgré le faible nombre de données disponibles, l'accroissement de la précision par notre processus est de l'ordre de 8% basé sur les données d'origine et de l'ordre de 5% sur les données interpolées.

Dans nos futurs travaux, l'ordre des pixels sera étudié plus précisément. Nous avions jusqu'à présent analysé les pixels d'une surface carrée dans laquelle le polygone était inclus, mais nous devons définir un ordre adapté à la géométrie du polygone lui-même. Nous allons également augmenter le nombre d'instances de vergers et appliquer la même approche à des problèmes impliquant un plus grand nombre de classes afin de générer des cartes d'occupation du sol.

## Remerciements

Ces travaux ont été financés par l'ANR sous le numéro de projet ANR-17-CE23-0015.

## Références

Andres, L., W. Salas, et D. Skole (1994). Fourier analysis of multi-temporal AVHRR data applied to a land cover classification. *Int. J. Remote Sens. 15*(5), 1115–1121.

Bagnall, A., J. Lines, A. Bostrom, J. Large, et E. Keogh (2017). The great time series classification bake off : a review and experimental evaluation of recent algorithmic advances. *DMKD 31*(3), 606–660.

Bruzzone, L. et D. Prieto (2000). Automatic analysis of the difference image for unsupervised change detection. *IEEE Transactions on Geoscience and Remote Sens. 38*(3), 1171–1182.

Butz, A. (1971). Alternative algorithm for Hilbert's space-filling curve. *IEEE Trans. on Computers 20*(4), 424–426.

Chelali, M., C. Kurtz, A. Puissant, et N. Vincent (2019). Urban land cover analysis from satellite image time series based on temporal stability. In *JURSE, Procs.*, pp. 1–4.

Chelali, M., C. Kurtz, A. Puissant, et N. Vincent (2020). Image time series classification based on a planar spatio-temporal data representation. In *VISAPP, Procs.*, pp. XX–XX.

Coppin, P., I. Jonckheere, K. Nackaerts, B. Muys, et E. Lambin (2004). Digital change detection methods in ecosystem monitoring : A review. *Int. J. Remote Sens.*, 1565–1596.

Di Mauro, N., A. Vergari, T. M. A. Basile, F. G. Ventola, et F. Esposito (2017). End-to-end learning of deep spatio-temporal representations for satellite image time series classification. In *DC@PKDD/ECML, Procs.*, pp. 1–8.

Huang, B., K. Lu, N. Audebert, A. Khalel, Y. Tarabalka, J. Malof, et A. Boulch (2018). Large-scale semantic classification : Outcome of the first year of inria aerial image labeling benchmark. In *IGARSS, Procs.*, pp. 6947–6950.

Iandola, F., M. Moskewicz, K. Ashraf, S. Han, W. Dally, et K. Keutzer (2016). Squeezenet : AlexNet-level accuracy with 50x fewer parameters and <1MB model size. *CoRR abs/1602.07360*.

Ienco, D., R. Gaetano, C. Dupaquier, et P. Maurel (2017). Land cover classification via multitemporal spatial data by deep recurrent neural networks. *IEEE Geosci. Remote Sens. Lett. 14*(10), 1685–1689.

Inglada, J., A. Vincent, M. Arias, B. Tardy, D. Morin, et I. Rodes (2017). Operational high resolution land cover map production at the country scale using satellite image time series. *Remote Sens. 9*(1), 95–108.

Ismail Fawaz, H., G. Forestier, J. Weber, L. Idoumghar, et P. Muller (2019). Deep learning for time series classification : A review. *DMKD 33*(4), 917–963.

Pelletier, C., G. Webb, et F. Petitjean (2019). Temporal convolutional neural network for the classification of satellite image time series. *Remote Sens. 11*(5), 523–534.

Petitjean, F., J. Inglada, et P. Gançarski (2012a). Satellite image time series analysis under time warping. *IEEE Transactions on Geoscience and Remote Sens. 50*(8), 3081–3095.

Petitjean, F., C. Kurtz, N. Passat, et P. Gançarski (2012b). Spatio-temporal reasoning for the classification of satellite image time series. *PRL 33*(13), 1805–1815.

Senf, C., P. Leitao, D. Pflugmacher, S. Van der Linden, et P. Hostert (2015). Mapping land cover in complex mediterranean landscapes using landsat : Improved classification accuracies from integrating multi-seasonal and synthetic imagery. *Remote Sens. Environ. 156*, 527–536.

Tran, D., L. Bourdev, R. Fergus, L. Torresani, et M. Paluri (2015). Learning spatiotemporal features with 3D convolutional networks. In *ICCV, Procs.*, pp. 4489–4497.

Verbesselt, J., R. Hyndman, G. Newnham, et D. Culvenor (2010). Detecting trend and seasonal changes in satellite image time series. *Remote Sens. Environ. 114*(1), 106–115.

## Summary

Image time series such as MRI functional sequences or Satellite Image Time Series (STIS) provide valuable information for the automatic analysis of complex patterns through time. A major issue when analyzing such data is to consider at the same time their temporal and spatial dimensions. In this article we present a novel data representation that makes image times series compatible with classical deep learning model, such as Convolutional Neural Networks (CNN). The proposed approach is based on a novel planar representation of image time series that converts $2D+t$ data as $2D$ images without loosing too much spatial or temporal information. Doing so, CNN can learn at the same time the parameters of $2D$ filters involving temporal and spatial knowledge. Preliminary results in the remote sensing domain highlight the ability of our approach to discriminate complex agricultural land-cover classes from a STIS.